Find the LCM and GCD of n numbers?

CODE:

import java.util.Scanner;

public class LCMGCD {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("N value: ");

int n = scanner.nextInt();

if (n < 1) {

System.out.println("Invalid input! N should be a positive integer.");

return;

}

int[] numbers = new int[n];

for (int i = 0; i < n; i++) {

System.out.print("Number " + (i + 1) + " = ");

numbers[i] = scanner.nextInt();

}

int lcm = findLCM(numbers);

int gcd = findGCD(numbers);

System.out.println("LCM = " + lcm);

System.out.println("GCD = " + gcd);

}

public static int findLCM(int[] numbers) {

int lcm = numbers[0];

for (int i = 1; i < numbers.length; i++) {

lcm = lcm \* numbers[i] / findGCD(lcm, numbers[i]);

}

return lcm;

}

public static int findGCD(int[] numbers) {

int gcd = numbers[0];

for (int i = 1; i < numbers.length; i++) {

gcd = findGCD(gcd, numbers[i]);

}

return gcd;

}

public static int findGCD(int a, int b) {

while (b != 0) {

int temp = b;

b = a % b;

a = temp;

}

return a;

}

}

OUTPUT:

C:\javap>javac LCMGCD.java

C:\javap>java LCMGCD

N value: 5

Number 1 = 7

Number 2 = 8

Number 3 = 9

Number 4 = 1

Number 5 = 2

LCM = 504

GCD = 1

